EXHIBIT F

Google AI with Jeff Dean

gcppodcast.com/post/episode-146-google-ai-with-jeff-dean

Jeff Dean, the lead of Google AI, is on the podcast this week to talk with Melanie and Mark about AI and machine learning research, his upcoming talk at Deep Learning Indaba and his educational pursuit of parallel processing and computer systems was how his career path got him into AI. We covered topics from his team's work with TPUs and TensorFlow, the impact computer vision and speech recognition is having on AI advancements and how simulations are being used to help advance science in areas like quantum chemistry. We also discussed his passion for the development of AI talent in the content of Africa and the opening of Google AI Ghana. It's a full episode where we cover a lot of ground. One piece of advice he left us with, "the way to do interesting things is to partner with people who know things you don't."

Listen for the end of the podcast where our colleague, Gabe Weiss, helps us answer the question of the week about how to get data from IoT core to display in real time on a web front end.

Jeff Dean

Jeff Dean joined Google in 1999 and is currently a Google Senior Fellow, leading Google AI and related research efforts. His teams are working on systems for speech recognition, computer vision, language understanding, and various other machine learning tasks. He has co-designed/implemented many generations of Google's crawling, indexing, and query serving systems, and co-designed/implemented major pieces of Google's initial advertising and AdSense for Content systems. He is also a co-designer and co-implementor of Google's distributed computing infrastructure, including the MapReduce, BigTable and Spanner systems, protocol buffers, the open-source TensorFlow system for machine learning, and a variety of internal and external libraries and developer tools.

Jeff received a Ph.D. in Computer Science from the University of Washington in 1996, working with Craig Chambers on whole-program optimization techniques for object-oriented languages. He received a B.S. in computer science & economics from the University of Minnesota in 1990. He is a member of the National Academy of Engineering, and of the American Academy of Arts and Sciences, a Fellow of the Association for Computing Machinery (ACM), a Fellow of the American Association for the Advancement of Sciences (AAAS), and a winner of the ACM Prize in Computing.

Cool things of the week

- Google Dataset Search is in beta site
- Expanding our Public Datasets for geospatial and ML-based analytics blog
  - Zip Code Tabulation Area (ZCTA) site
- Google AI and Kaggle Inclusive Images Challenge site
- We are rated in the top 100 technology podcasts on iTunes site
- What makes TPUs fine-tuned for deep learning? blog

Interview

- Jeff Dean on Google AI profile
- Deep Learning Indaba site
- Google AI site
- Google AI in Ghana blog
- Google Brain site
- Google Cloud site
- DeepMind site
- Cloud TPU site
- Google I/O Effective ML with Cloud TPUs video
- Liquid cooling system article
- DAWNBench Results site
- Waymo (Alphabet's Autonomous Car) site
- DeepMind AlphaGo site
- Open AI Dota 2 blog
- Moustapha Cisse profile
- Sanjay Ghemawat profile
- Neural Information Processing Systems Conference site
- Previous Podcasts
  - GCP Podcast Episode 117: Cloud AI with Dr. Fei-Fei Li podcast
  - GCP Podcast Episode 136: Robotics, Navigation, and Reinforcement Learning with Raia Hadsell podcast
  - TWiML & AI Systems and Software for ML at Scale with Jeff Dean podcast

- Additional Resources
  - arXiv.org site
  - Chris Olah blog
  - Distill Journal site
  - Google's Machine Learning Crash Course site
  - Deep Learning by Ian Goodfellow, Yoshua Bengio and Aaron Courville book and site
  - NAE Grand Challenges for Engineering site
  - Machine Learning for Systems and Systems for Machine Learning slides

Question of the week

How do I get data from IoT core to display in real time on a web front end?

Where can you find us next?

Melanie is at Deep Learning Indaba and Mark is at Tokyo NEXT. We'll both be at Strangeloop end of the month.

Gabe will be at Cloud Next London and the IoT World Congress.

Transcript

hide full transcript

[MUSIC PLAYING] MARK: Hi, and welcome to episode number 146 of the weekly Google Cloud Platform podcast. My name is Mark Mandel. And I'm here as always with my colleague, Melanie Warrick. Melanie, how are you doing?

MELANIE: Hi, Mark. Doing great. How are you doing?

MARK: I'm doing all right. I'm doing very well, very excited for today's episode.

MELANIE: Yes, this week in particular we're very excited to have with us Jeff Dean, who you may have heard about him if you know a little bit about Google and AI and TensorFlow, and TPUs, and big data systems, and cats.

[PURRING]

You may have heard about him.

MARK: Cats.

MELANIE: Cats. Anyway, all the things.

MARK: It's a thing. OK, cool.

MELANIE: All the things that come to computer systems, hardware, and all that. So we had a great conversation with him and covered a lot of these topics.

But part of the reason why we have this interview this week is because this week is Deep Learning Indaba, which is happening out in Stellenbosch in South Africa. And it's bringing together researchers from different countries within the African continent to share their knowledge. Jeff Dean is going to be speaking there, and I'm also going to be there recording some episodes.

So we wanted to emphasize that a little bit. We don't actually get into the podcast about it until later in, but you can listen and hear about what he's going to talk about at the conference.

As always, before we get into that, we are going to start out with our cool things of the week. And we will end with the question of the week. And we have a special guest joining us to help with the question of the week. How do I get data from IoT Core to display in real time on a web front end?

MARK: Yes, and we have Gabe Weiss joining us today.

GABE: Hi, friends.

MELANIE: So to start out the cool things of the week, something that I caught was that Google has recently put into beta Dataset Search. And so I caught this actually because Ben Hammer, who is the CTO at Kaggle, had tweeted this out today. And specifically, he was noting how, you know, this helps to index metadata on open datasets and should be pretty useful for people who are researchers and those that are curious about public datasets and wanting to explore them.

So we'll include the link to it, but it's pretty much a search tool. It's a Google search tool. It's just specifically on datasets.

MARK: Awesome. I wanted to talk about this really cool thing I like, too, which is we're expanding our public datasets for geospatial and ML-based analytics. So you may have heard back at Cloud Next, we announced an additional five petabytes of BigQuery storage available for public datasets. In the blog post we'll link to in the show notes, we also talk about that this additional storage will also be available for the next five years, which is also really cool.

Since Next, though, we have onboarded seven new datasets that define boundaries in the United States by parameters such as ZIP code and things like that, basically to support geospatial queries. So we have a whole bunch of new data in there for you to do to geospatial stuff, to do ML-based analytics, to use the new ML toolkits that are available now in BigQuery.

And if you want take advantage of any of those, make sure you head over to the Google Cloud Platform Marketplace. There's a datasets filter you can go into, and at last look, it looks like we have about 101 different public datasets for you to play with on BigQuery. And you can do up to a terabyte of querying on BigQuery for free, which is also pretty amazing, per month.

MELANIE: Another thing we wanted to mention again, referencing Kaggle and also Google AI, is that there is a new challenge through Kaggle Google AI has posted, specifically to encourage the users and the Kagglers out there to develop models that are more robust to blindspots that exist in datasets. We've talked about this in the past. I've seen many people discuss machine learning bias and fairness in particular. And that really stems from the data itself and how the data can really be biased in what you've gathered and how that can then influence the model that's built.

So their challenge is specifically to help identify techniques and approaches to account for that and to make the model more robustness. This challenge, in particular, is set to run in two stages. They've got the information on the website that we'll keep a link into, but it's going to wrap up around the time of NIPS, that's later this year, December. And so you can check that out and experiment.

MARK: Excellent. I also want to mention that we were checking out the top 100 podcasts in tech recently on iTunes, and we're in it--

[CELEBRATORY HORNS]

--which is kind of cool. So thank you very much to everyone who's been listening, people who have rated us on iTunes, and basically have supported the podcast. That's a really nice, little thing.

MELANIE: Yes. And last but not least, if you want to know what makes TPUs fine-tuned for deep learning, we've got a blog post that we're going to share that you can look into this. And we figured this is especially relevant for today's episode.

But it helps step you through what neural nets are. It helps step you through how TPUs work, as well as in comparison to CPUs. And you see these nice graphics that do a good job in terms of stepping you through what that looks like visually.

MARK: Nice.

MELANIE: And some pricing information at the bottom, too.

MARK: Very cool.

MELANIE: All right. Mark, I think we should go ahead and get into it. We've got a great interview now with Jeff.

MARK: Let's go talk to your buddy Jeff.

MELANIE: Yeah, my buddy. We play cards.

MARK: [CHUCKLES]

MELANIE: We're thrilled today to have with us Jeff Dean, who is heading up Google AI. Thank you, Jeff, for joining us.

JEFF: Thanks very much for having me.

MELANIE: So Jeff, I mean, I know a lot of people know who you are. But we always ask all of the people who come on the show to tell a little bit about themselves, what your background is, what you do.

JEFF: Sure. So at the moment, I am leading Google's AI division. Internally, we know that as Google Research and Mission Intelligence. But externally, we kind of use Google AI sometimes, depending on who we're talking to.

And so that organization does lots of different kinds of computer science research. We do fundamental research trying to create new algorithms, new techniques to solve problems. We do a fair amount of work with Google product areas, to work with interesting research problems in the context of products and get, sort of, our research out into the Google products that everyone knows and loves and uses.

We also have been doing a bit more work on, sort of, systems infrastructure for enabling our research, but then also creating tools that enable internal developers at Google but also external developers when we open source things. So TensorFlow for example, came out of our group and is a popular machine-learning toolkit. We feel like that's a good way to really broaden the impact of the research that we're doing is to make it easy for people to then build on the work that we're doing and have good tools available.

And then the fourth kind of thing that we generally do is try to find new areas that Google doesn't currently work in where we think our research can make a big impact. So we've been doing a lot of work on, sort of, machine learning for health care, machine learning for robotics, because we think those are kind of new emerging areas where machine learning and our research will make a big difference.

MELANIE: In terms of the work that your team is doing, what does it mean for you? And what does it look like for you now that-- because I know this is a fairly new position, to a degree, that you're in as the leader of Google AI this year, right?

JEFF: Yes. Yes, I took on this role in, I guess, late March, early April. And basically my goal is to inspire our great researchers to do great research and to tackle the important, ambitious problems that I think we should be working on. I spend Mondays coding--

MELANIE: Oh, wow.

JEFF: --and, with my colleague Sanjay Ghemawat, who I've been working with for many years. I, kind of, probably have four or five other projects that I'm involved in, in an actual technical level, where I go to the weekly meetings and make sort of technical suggestions and

so on.

MARK: Mm-hm.

JEFF: And then the rest of my time is spent kind of trying to steer a reasonably large organization to do the right things and to make sure that we're focused on the right problems and have impact.

MARK: Awesome.

MELANIE: I'm curious, what do you code in?

JEFF: Mostly C++.

MARK: Oh, fun. So I like asking these questions for the people who work in AI and ML because we get varying answers. What does AI mean to you?

JEFF: It means basically being able to have computer systems that can solve problems that you would think of as requiring human intelligence.

MARK: I think that's actually the most concise answer we've ever got.

MELANIE: Yeah, pretty straightforward. Well, in terms of the work that you do, what do you think are some of the biggest challenges that you are up against?

JEFF: Really, it's an exciting time to be in the field of computer science research and machine learning research because there is an explosion of interest in this field, as in the last maybe seven or eight years, collectively as a research community, we really pushed forward what computers can do. I think this is the, sort of, most striking in the computer-vision field, where essentially using deep-learning models for computer vision has suddenly transformed computers from not being able to see very well to now actually being able to see.

And if you think about the impact on the world that that will have, and is already having, that's transformative. Suddenly you can have automated machines able to perceive the world around them, able to sort of understand what they're looking at. And that has huge implications for, you know, general computer vision.

So it's a key enabler for a lot of the experience that Google Photos provides to Google users. Like we can understand that's a picture of a Doberman and that's a mountain. But it also means transformative things in health care. So tons and tons of medical imaging-related problems now are sort of tackleable with automated machine-learning algorithms and assisting sort of medical professionals in that way.

MELANIE: Right.

JEFF: Robotics, it's very clear, if you want to build a robot, it's very helpful if the robot can see.

MELANIE: Yes.

JEFF: So that has big implications. And I think we're making significant progress in other fields related to machine learning. So things like natural language understanding and speech recognition, you've seen tremendous advances over the last six or seven years. And that really means there's all these opportunities about how we should be using these new capabilities to affect the world--

MELANIE: Right.

JEFF: --to make it a better place to now build new things that we couldn't build before. And I think that's the excitement and the flood of interest in studying this field. How do people do research in this field? How can more people enter this field and do research?

MELANIE: So we spoke earlier this year with Dr. Fei-Fei Li around Cloud AI. We spoke not too long ago with Raia Hadsell out of DeepMind and talked about some of the robotics research that they're doing. How does Brain and Google AI collaborate with the different groups?

JEFF: Right. So we actually have strong collaboration across all of Google and even across all of Alphabet in many areas. So with Cloud AI, for example, many of the, sort of, things that they are bringing to market started as research projects in the Google AI Research division.

And we're now collaborating closely to bring things like the AutoML research that was started into products like Cloud AutoML, where now customers who have computer vision or other kinds of problems that maybe don't have sophisticated machine-learning developers on staff but want to be able to take advantage of the machine-learning capabilities that now exist, we have systems that can automatically train and learn to solve new problems that a customer might have. They might have pictures of broken parts on their assembly line and not broken parts, and they want to be able to distinguish. And they can essentially just upload those images and get a trained model that helps them with that particular task, even without having sort of a master's level, machine-learning expert on staff.

MELANIE: Makes sense. I know one of your areas of passion, in particular and where you've had a significant impact especially, is around the tensor processing units, the TPUs. What helped drive or inspire the initial development of TPUs? And I know you've had this conversation before, but you know, especially from your experience, as you were coming along and getting into technology to begin with, how did that come about for you?

JEFF: Right. So I've always been interested in, how can computers solve interesting problems? And one interesting way of doing that is to bring more computation to bear on problems that you can then use more sophisticated algorithms. Or you can use a larger

datasets to derive better insights.

And so having more computation is generally a good thing in computing. And around the time of maybe 2011, 2012, when the Google Brain Project that I co-founded was just getting started, we started to collaborate with a number of different product teams at Google to use deep learning in some of the products.

And the ones we collaborated with most closely with were the speech recognition team to replace kind of the older style machine-learning model that they were using for speech recognition with a deep learning-based model for the acoustic portion of it, the part of the model that goes from a very small audio recording to a part of a word. Is it a "buh" or "fff" or a "sss"?

MELANIE: Yeah.

JEFF: And then there was another model that we didn't focus on initially, kind of after that state of the processing. But replacing the acoustic model with a deep learning-based acoustic model gave a lot of gains in recognition accuracy. And so we could tell that as speech recognition gets better, people are going to use it more and more.

And so I started to do some "back of the envelope" calculations of, like, well, what happens if people start talking to their phone three minutes a day, right? Because they're going to draft all their emails, a speech, or something. And at the time, we had, you know, just lots and lots of CPUs in our data center.

And if you looked at how much computation that would be required if 100 million of our users started to do that, that was actually kind of daunting and scary. We would have to essentially double the computing footprint of Google just to support, like, a slightly better speech recognition model for modest fraction of our users.

MELANIE: Seems legit.

JEFF: Yeah. So it seems a little scary. But it's clear we want to deploy that to our users because the gains in recognition accuracy are actually quite significant. And so we started to look at what could we do for these kinds of deep-learning models that would be more computationally efficient. And there are two really nice properties that deep-learning models have.

So first, they are very tolerant of reduced precision. So essentially, if you do that precisions to like one decimal digit of accuracy, that's actually perfectly fine for these models. You don't need six or seven digits of precision like you would in floating point computations or even more in double computations.

And the second property they have is that all the algorithms are made up of, like, different compositions of handful of building blocks, essentially matrix multiply, vector dot products, linear algebra-style operations. So if you can build hardware that is only designed to accelerate low-precision linear algebra, you're golden.

And that enables you to then really tailor the hardware to do only that. It doesn't need to do twisty, branchy, C++ code of all the kinds of arbitrary things, pointer dereferences, just low-precision linear algebra. Then suddenly you can rethink how you would completely design a computer to do only that.

MARK: Since it's focused in exactly what it is it can do, does that mean there are certain applications that are better or worse for a TPU that you might use something else for? Or how does that work?

JEFF: Right. So this is one of the double-edged swords of specialization is if you over-specialize, then it's not generally applicable. But if you make something very generally applicable, then it doesn't get all the performance benefits you could get from specialization.

And we actually chose, I think, a pretty good balance for TPUs. They're essentially designed to accelerate exactly the kinds of operations you find in deep-learning models. So they're not general purpose sort of computational devices. But they are not tailored to a particular model.

They are tailored to all the kinds of operations you generally find in deep-learning models. And so you can run vision models on them. You can run speech models. You can run recurrent language models.

The first TPU, TPU v1, was targeted at only inference, where you actually need even less precision than the training process. And that was our most pressing problem. That was how we could get the speech recognition model out to the world without doubling Google's data center footprint.

And then the second TPU was really both-- was targeted at both training and inference. And the reason we didn't bite that off first was that training is a much more complex hardware design problem, because for inference, usually you can design things so that a single model will fit on one chip. And if you need more capacity, you just kind of stamp out lots of copies of that system and put a lot of boards and a lot of computers, and all of a sudden you have lots of speech recognition capacity.

For training, it's very unlikely for large models you can get a single chip to be fast enough to train a model, like, very fast. And what you really want in training is fast turnaround for machine-learning experiments or huge scalability, so you can train very large models on very large datasets very quickly and be able to iterate from a machine-learning research perspective. Or a deployed system, you might want to retrain your model every 10 minutes because your data changes.

There, it's not just a chip-design problem. It's actually a whole computer system. It's more like a supercomputer design problem, where you have the chips, and then you have some complicated interconnect that connects them together, or you have cooling issues for how do you cool this giant thing. It's made up of multiple racks. And so that's why we didn't bite that off in the first system but chose to wait till the second iteration of the TPU design to tackle training as well.

MELANIE: And you announced this year version 3 of TPUs as well.

JEFF: Yes.

MELANIE: And am I right in remembering that the cooling system, it's now liquid?

JEFF: Yeah. So it turns out-- yeah, yeah, yeah, it is fancy-- any time you mix computers and water it's always exciting.

[LAUGHTER]

So the TPU v3, which Sundar announced it at Google I/O in May 2018, essentially got water cooling. So a board has four of these chips on it, and the water cooling kind of goes to the surface of these chips and takes excess heat away.

And the largest footprint deployment, we call these things pods, the TPU v2 pods were essentially made up of 64 of those devices, 256 TPU v2 chips. And the pod scale for the TPU v3 is much larger. So we have about eight times as much computation in the TPU v3 pod as we did in the TPU v2 pod.

MELANIE: I know a couple of the people on your team was specifically running a Stanford DAWNbench on it, and they were showing the differences in terms of its performance. Like right now, all that's going in my head right now is the ImageNet. But--

JEFF: I can do it.

MELANIE: Please.

JEFF: So DAWNbench is a new machine-learning benchmark set up by a research group at Stanford, the DAWN research group, that is designed to look at a variety of different kinds of machine-learning problems, both training and inference, and measures a few different metrics. So one of them is for a particular machine-learning problem, how quickly can you get to a certain level of accuracy that is sort of a good level of accuracy for that problem? And one of them is a ImageNet processing benchmark, where you have to get to 76% Top-1 accuracy.

MELANIE: And this is classifying images.

JEFF: Classifying images.

MELANIE: ImageNet is very much like that.

JEFF: Right.

MELANIE: This is what this is--

JEFF: You got a color image. You have to classify it into one of 1,000 categories. It's actually pretty hard. Human error rate in Top-1 one accuracy is about 5%.

MARK: Oh, wow.

JEFF: So because it's got like 40 breeds of dogs, and you have to be able to distinguish.

MARK: And so I'm guessing the TPUs did pretty OK.

JEFF: Yeah, so the TPUs took the top three spots in the overall time metric with pods, and then also the top two spots, I believe, in the cost using public cloud computing resources.

MELANIE: Cool. Well, what do you enjoy most about working with these chips and working with parallel processing?

JEFF: I actually have been excited about parallel processing for quite some time, ever since I took a course my senior year at University of Minnesota on parallel processing. And at the time, I was really excited about that because I felt like more computation was the answer to solving a lot more problems. And this was kind of during the first wave of excitement about neural nets that happened in the late 80s and early 90s, when new algorithms were developed and it was first surfacing as a potentially interesting way to tackle some difficult problems.

And at that time, basically neural nets could get interesting results on very toy-sized problems but not, sort of, scale to really interesting problems. And so there was a wave of excitement about this, but then in this wave of excitement, I kind of felt like more computation was going to be helpful for letting us tackle bigger problems. And so I did an undergrad thesis, working with the professor who taught the parallel and distributed computing class I took, sort of did an independent thesis on parallel training of neural nets.

And so I felt like if we could take the 64 processor machine we had in the department and apply parallel methods to it, we could get, you know, a factor of 60 more compute applied to these things. And then we could scale up and tackle bigger problems. In retrospect, we needed like a million times more computation, not 60.

And so if you essentially just wait it out with Moore's law, at least until about 5 or 10 years ago when Moore's law significantly slowed down, but before that, the previous 25 years, 40 years even, we were getting these consistent, you know, 2x performance improvement every two years. You just wait long enough, and all of a sudden voila, a million times as much compute, and you're golden.

And that's sort of what started to happen in like 2008, 2009, is we suddenly had enough compute, initially using GPU cards, which were designed for gaming, but obviously have turned out to be quite useful for training deep-learning models, because a lot of the acceleration you want to do for graphics processing turns out to be amenable to training deep-learning models. And so that was when people in the academic community started to really see that we were starting to get interesting results on real problems using neural nets.

MELANIE: Yeah.

JEFF: So there was kind of excitement started to build back up.

MELANIE: What was one of the hardest parts in your career? It wasn't like AI was well accepted for many, many years when you were studying, between then and 2006, really, right? So what was some of the hardest moments in terms of pursuing this interest, pursuing this research?

JEFF: Yeah, I actually kind of come at this as a general computer systems person. So I like to view-- the kind of lens I bring to problems is there's an interesting problem. Let's figure out how we can build a computer system to tackle that problem, often involving, you know, either kind of good abstractions for expressing computations we want to perform or using large amounts of computation and figuring out how to paralyze or distribute computation across lots of machines, so that we can scale up the competition.

So although I was doing work as an undergrad on this parallel training of neural nets, I was coming at it from the perspective of, how can we build a computer system to do this? And then I sort of started to get back interested in machine learning in, like, 2011, when I happened to bump into Andrew Ng in a micro-kitchen--

MELANIE: I was going to say, the story goes you are in a micro-kitchen, right?

JEFF: Yes. I knew him a little bit. And so I said, oh, what are you doing here? He's like, oh, I'm spending a day a week at Google X, and I haven't quite figured out what I'm doing here yet. But, you know, my students at Stanford are starting to, like, see what kinds of problems can be solved with neural nets, and they're getting pretty good results.

I'm like, oh, really? That's pretty cool. I did a bit of work on neural nets as an undergrad. I said, they're really working now? And he's like, yeah. So I said, we should train really big neural nets.

And so that was sort of the genesis of how we decided to form the Google Brain Project along with Greg Corrado. And we basically, at that time, didn't have accelerators or data centers. But we said, we have a lot of computers. Let's figure out how to make a computer system that can train really big neural nets--

MELANIE: Let's leverage this.

JEFF: --using CPUs.

MARK: And so, like, is that the moment where you were like, this is going to be a really big thing? Like did you think it then?

JEFF: I did feel like once we kind of did a little bit more work, maybe another month of dabbling, then it seemed like, hey, we can actually tackle problems using neural nets-- that people have felt like they're the right abstraction for a long time, but maybe they would suddenly start to work in real problems if we could apply lots of computation. And so that's really been what a lot of my focus has been on the last six or seven years is how can we bring lots of computation to the problems that you want to tackle with machine learning, and in particular deep learning?

MELANIE: So we're recording this the day after the Neural Information Processing Systems Conference sold out--

MARK: Yes.

MELANIE: --in like 11 minutes and so many seconds.

MARK: 48, I think.

MELANIE: 48 seconds.

MARK: 11 minutes and 48 seconds.

MELANIE: Using this as an example, there's a lot of excitement, clearly, in this field. How do you feel about all this excitement?

JEFF: You know, there is a fair amount of hype. But on the other hand, the hype, some of it, is actually justified because we're now actually able to build systems that can solve problems we couldn't solve five or 10 years ago.

I mentioned computer vision. Speech recognition has advanced tremendously. Translation has advanced. Many applications of these sorts of things have shown real improvements or new capabilities in what computers can do.

So I think that's part of why people are excited. Many, many people want to enter the field. So you see enrollment in machine-learning classes, undergraduate and graduate levels, just going through the roof.

You see more and more people attending the major machine-learning conferences. You see the number of research papers on archive actually growing at a faster exponential rate than Moore's law over the last seven or eight years. It's quite remarkable.

And, you know, I think it's important to temper people's expectations who think we're going to have super intelligent things that can solve problems, general problems that humans can't solve, in the next, you know, six months. That's kind of the one level of hype that's clearly not justified.

But it's also important to recognize that computers can now see. That's a pretty big deal. And that has major implications across a bunch of different areas.

You know Google and Alphabet are working on an autonomous cars. Clearly computer vision and perception is an important component of making safe autonomous cars. Robotics, machine learning for medical imaging, machine learning for other medical applications, health care applications, these are going to transform major problems in the world. And that's good. That's why there's hype.

It's important to tamp down excessive hype, but not to say that there shouldn't be any hype, because I think we are making significant advances in what computers can do.

MELANIE: Any application in particular that you've seen that you feel like don't get a lot of attention but are pretty impactful?

JEFF: Yeah, I mean, I think there's a realization that I've come to from looking at a few different problems researchers in our group have tackled and other work that's going on, where in many scientific problems, there are, for example, traditional, high-performance computing-based simulations of some property, some scientific thing, maybe you're simulating chemical interactions at a very fine-grain level, how do the electrons flow around? And what properties does that have?

Or maybe you're simulating earthquake faults, and you have a very detailed, sort of, physics-based model of how slippage in earthquake faults happens. And often, these computational models are very, very expensive computationally. So you run one chemical through your simulator, and an hour later, you get information about the quantum properties of this chemical, or what binds to light, or things like that.

And you can actually use that simulator as a teacher. And so if you have a really expensive scientific simulator, often you can train a neural net to do what that simulator does, completely end to end. You take the inputs you would have fed to the simulator, and you get the outputs. And you run a bunch of examples through the simulator. Now you have a training set for a neural net.

And in a few different domains, we've actually seen people do this. And in, for example, in the quantum chemistry domain, George Dahl and others in our research group worked with some chemists at different universities and took the simulator they were using, trained a neural net to do it. And now they can't distinguish the accuracy versus the original, sort of, HPC-style simulator, but the tool is 300,000 times faster.

MARK: Wow.

JEFF: And any time you take your tool, and you make it 300,000 times faster, that just transforms how you would do science. Right? You can imagine, I'm going to go make coffee. Let me screen 100 million molecules, and then when I come back, I'll look at the 10,000 that are most interesting.

And that same thing happened with Brendan Meade, who is a visiting Harvard faculty member who visited our group in Cambridge for a year and did a bunch of work on earthquake fault simulation. He relayed that in work he'd just done before he came to visit us that he'd replaced the earthquake fault simulation inner loop with, kind of, the world's lamest neural net. It was, you know, four layers of 10 neurons each. And all of a sudden, the thing was like between 10 and 100,000 times faster.

MELANIE: Oh, wow.

JEFF: And you couldn't tell the difference in accuracy.

MARK: Wow.

JEFF: So I think that's a good lens to look at, are you doing scientific computing or simulation of some sort? Can you then train a neural net to do an approximation of that that is quite likely to be accurate if you're given enough training data?

MARK: Now I'm just thinking of--

MELANIE: Giving you ideas?

MARK: Well, I work with games, and so I'm like if you can get that kind of performance for that level of simulation you can do some really interesting things.

JEFF: Right. Yeah.

MELANIE: And I think I saw you speak about this, too, that you're using neural nets as well to help with improving the actual performance of the systems themselves.

JEFF: There's a big opportunity in general computer systems. So if you think about what's at the heart of operating systems, or compilers, or storage systems, or other computer systems, you know they're generally laden with kind of handwritten heuristics. You know, which process should the operating system schedule next out of the ready processes that are available? Or for a compiler, should it, you know, emit code with this LoopNet ordering or this loop ordering? Or which variable should it spill to memory when it can't fit them all in registers?

And in many cases, I believe these are actually learning problems. And so the heuristics that people tend to handwrite have to work well in the general case. And they can't really be developed in such a way that they completely adapt to how the system is being used.

So to give you an example, in our Bigtable storage system, you know there's essentially a key value store. And client requests come in to Bigtable servers, and they can request that certain data be read from different tables. And when that happens, the Bigtable server decides to put it in an in-memory cache so that if data is needed again, then you have it in the cache. You don't have to go to disk and fetch it.

And if you look, lots of information is available at the time you make the decision to read the data from disk. We know, for example, the internal user group of the job that's running, the process that requested the data. We know maybe the job name that requested this.

And you would never write a heuristic. So a lot of the processing is from sequential MapReduce-style processing jobs that are going to read the data once, never use it again, until like an hour later, when a similar MapReduce starts up. But you don't really want to insert that in the cache, because you're not going to get any reuse, sort of, immediately.

But you would never write a heuristic in the middle of your Bigtable server that says, you know, if job name starts with MapReduce DASH, then, you know, don't insert in the cache. But if you look at what a learning heuristic could do, it would pretty quickly learn the pattern that these jobs that seem to start with MapReduce DASH never seem to reuse the data I inserted in the cache. So I'm just not going to insert it.

And I think there's huge potential for the thousands, and thousands, and thousands of heuristics in computer systems to really mostly be learned and adapt to how they're actually being used.

MELANIE: That's wonderful and exciting, too, to think about the ways that this can be improved upon. I feel like there are so many things I want to ask you, and we definitely are going to be limited on time. So I want to make sure I make some space and time for the fact that you're going to be in South Africa when we release this. And specifically, you're there to speak at a Deep Learning Indaba that's taking place outside of Cape Town. So can you tell us a little bit about how you're involved in this, the conference itself and what you're going to be speaking about?

JEFF: Sure. So I'm going to give a talk on solving challenging problems in the world of deep learning, or something.

MARK: Nice.

JEFF: And actually the way it's framed is around the National Academy of Engineering put out in 2008 a list of important problems for the engineering community writ large to work on in the 21st century. And it's like a list of 14 different areas, things like improve medical

informatics, develop better drugs, improve solar energy output. And if you look at them, I actually think machine learning will be useful for tackling smaller or larger pieces of all of those different things.

MELANIE: Yeah.

JEFF: So the talk is framed a bit as like, hey, here's some interesting work that we've done that is part of the way toward solving some of these grand challenge kind of areas.

MELANIE: Nice.

JEFF: And so the Deep Learning Indaba, I think, is a really exciting event. It's a gathering of, I think, it's about 400 or 500 people from around mostly the African continent, from different countries. And really there's a burgeoning and exciting community in Africa that is excited about using machine learning, and doing machine-learning research, using machine learning to tackle problems in countries in Africa and around the world.

And I think this is really exciting to see this swelling of a research community in Africa, as well as the rest of the world. So I'm excited to go take part in that. I actually lived in Uganda and Somalia as a kid.

MELANIE: Oh, wow.

MARK: Cool.

JEFF: So I just care deeply about--

MELANIE: It matters a lot to you to be able to participate and to see and grow this group.

MARK: Fantastic.

MELANIE: I know, too, that Brain has expanded into Ghana. And there's an outfit that's being led by Moustapha Cisse, who we, at some point, hopefully we'll get to come on the podcast as well because I've met him and he's wonderful. I know. He's great. And I know they're off the ground and running as we speak. Are there other groups that you're looking to grow, or other offices that you're looking to spin up that you can even speak about? Or is that something we can't talk about right now?

JEFF: We're very excited about the Ghana office, like we're in the final throes of renovating the office space, and we've hired a bunch of people for the office that I think is going to be a great addition to our global research organization. They're going to do fantastic work in both basic machine-learning research, looking at some, sort of, interesting applied problems that are particular to Ghana and the rest of Africa.

I think it's going to be great. I love Moustapha. He's a wonderful leader. And it's great that we have him in the office.

MELANIE: He's a wonderful person out there. The other thing I wanted to ask you, a couple other things. So looking forward to Deep Learning Indaba, anybody who's out there who has interest, there'll be resources we'll provide on the show notes.

The last couple things I want to make sure we touch on before we let you go, love to know what is one of the best pieces of advice that you've received in your career?

JEFF: I don't know if it's a specific piece of advice that I've received, but an observation I've made throughout my career is that really good way to do interesting things is to partner with people who know things that you don't. And I often find that working on projects with, like, three or four people, where you each have kind of different kinds of expertise, and collectively you kind of come together to tackle some interesting problem that none of you could solve individually, but collectively you can, because you have the full totality of techniques or can develop them.

And the reason that's really important is it's a good way to continuously learn new things, even after you get out of school. I think when you're in undergraduate, or graduate school, or whatever, you're obviously learning things, but it's important to continue to learn things and pick up new ideas and, sort of, stimulate your brain. And this is a really good way to do that.

And then you kind of go off with that project, and you go on to a different project, and some of the other people's expertise has kind of rubbed off as a sheen on you. And you can now at least look at problems through the lens of the kinds of expertise they have, and say, oh, yeah. That looks a lot like this. I know the questions to ask or the right approach to take in tackling this, even if that's not your, sort of, formal background.

MELANIE: Yeah, definitely. You had mentioned about Moore's law and the amount of research that's coming out and papers in particular. How do you keep up on the resources and the research and all that?

JEFF: Yeah. I mean, again, I think this is something where surrounding yourself with people who know things that you don't is a really good way, because if you have a good network of people that you interact with, they can partly be your eyes and ears and look out for interesting things in their area of focus, and say, oh, yeah. That's kind of interesting. Someone told me about this today.

And you can kind of start to connect the dots between different areas and see the potential for a technique developed over here, how it might be used over here, or how it might combine well with some other technique and really build on each other. So I think it's very hard to keep up with everything. So you need to, sort of, collectively find some things you really do follow yourself and then get a cursory level of information about other things.

One thing I've observed is that a lot of people really, really deeply read a research paper. And I think that's important sometimes, but it's kind of the wrong approach for general learning. I think you would be much better served to take the time to read 100 abstracts of 100 different

papers and sort of understand the highest level concept that they each have.

Because you can always go read that thing in more detail if you need to find out more about it. But being able to have those 100 pieces of information of like, oh, yeah. That's a technique. That's a thing. We can kind of do something like that. That's how you really start to connect the dots and make connections across fields or across different problems.

MARK: Nice.

MELANIE: Any specific public research you've heard of very recently that you're, like, specifically it comes to your mind that you're like, this is exciting to me? Or--

JEFF: Well, I think the general idea of having computers that can automatically do experiments and then measure the results of those, and then sort of continuously absorb the results of a first set of experiments, and then generate new experiments. So all of the neural architecture search work, or the AutoML work, that our group has been doing, a lot of the reinforcement learning-based work that is showing promise in solving games like Go, or OpenAI work on Dota.

I think all these things kind of point at that general direction of having computers that can automatically explore different approaches to solving problems is going to be important, because a computer can run 50,000 experiments, whereas a human runs 50, looks at the results, and then figures out what are the next 50 to run.

MARK: Yup.

MELANIE: OK. And then the other question I want to ask you, out of all the interviews you've ever done, what question do you wish people would ask you?

MARK: Ah.

JEFF: I will tell you one interesting thing that some people don't know. So I actually moved around a lot as a kid. I went to 11 schools in 12 years. And I think that--

MELANIE: That's a lot of schools.

JEFF: That is a lot of schools, yes. Like, oh, new year, new school.

MELANIE: You learn how to make a lot of friends, I'm sure.

JEFF: Yeah, I mean, I think it was pretty helpful for me, because it let me live in lots of different places, lots of different kinds of environments, everywhere from Hawaii, Boston, Uganda, Boston, Arkansas, Hawaii, Minnesota, Somalia, Atlanta, Geneva, Seattle, and the Bay Area. And--

MELANIE: You must not be open-minded at all.

[LAUGHTER]

JEFF: I think it does help, because you can find really great aspects of any of those places. And I think that's a good perspective to have.

MARK: Fantastic. Before we wrap up, any resources you want to recommend to our listeners for people who wan to get into deep learning, TPUs, Google Brain, just ML research?

JEFF: You know, I'm asked this question a lot, and a lot of it depends on the kind of level of expertise someone already has. So I think, I don't have a single answer, but I recommend, you know, there's lots of good blogs out there. Chris Olah's blog and the Distill.pub journal that he started with Shan Carter are really good expositions for people with a certain level of expertise.

I think there's things like the machine-learning crash course that Google put out. There's lots of courses on Udacity and Coursera for different levels of expertise. The deep learning textbook that Ian Goodfellow, Yoshua Bengio, and Aaron Courville put out is a very good one.

There's lots of material targeted at, sort of, people with less advanced math skills, for like high school students and middle school students even. So there's just a wealth of information out there. And it's really great that the community is producing so much introductory material and more advanced material for everyone.

MARK: Excellent.

MELANIE: Great, well, we did cover the fact that you're speaking at Deep Learning Indaba. Anything else that you wanted to mention before we let you go?

JEFF: No, I'm just really excited. Thanks for having me, and thanks for the conversation.

MELANIE: Thank you so much for coming on. We appreciate it.

JEFF: Sure.

MELANIE: Thank you, again Jeff. It was really great just to get into all the things. And one of the many things that I really appreciated from that was hearing that he still codes.

MARK: Yeah.

MELANIE: Like he's like, this is my day.

MARK: Monday, I code. Yeah.

MELANIE: That is when I will code. So it's hard to do that as you become more senior to make that type of space in your life. So anyways, thank you

MARK: Yeah.

MELANIE: All right. Question of the week.

MARK: So we have our wonderful friend here, Gabe.

GABE: [INAUDIBLE].

MARK: Hey. So Gabe, we have a wonderful question for you because we know you do stuff in IoT Core.

GABE: I'm ready.

MARK: If I have an IoT Core thing, how do I get my data from IoT Core to display in real time on a web front end? I believe you've done something like this.

GABE: I have, recently. So anyone that's worked in IoT knows there's a bajillion different ways to do this. There's different classes of devices. There's different protocols. There's different everything. So there's-- how many grains of sand on a beach are there? That's how many ways there are to do this.

The way that we did it as kind of our golden path at Cloud Next in 2018, we built an entire end-to-end app on stage. There were five of us. So we went from devices. We used Raspberry Pi with some sensors that we did measuring heart rate. And that will communicate with MQTT is the protocol that we used that IoT Core understands how to receive. So you can send your telemetry data from your devices to IoT core. From IoT Core, it gets bridged. All of your telemetry data becomes events in Cloud Pub/Sub, which is our event stream manager. From Cloud Pub/Sub, you can trigger Cloud Functions, which we did to pull the telemetry data out of Pub/Sub and put it into our data warehouse, which was Cloud Firestore. We picked Firestore because it's a NoSQL database. It's super low latency. And it has a special sauce, which is it has a push mechanism, so that front end apps can register for a callback when any data changes in the database. So for real-time applications, this is huge. So our f

ront end was an angular app that we wrote that subscribed to this callback from Firestore that pulls it in and draws our heart rates on a graph. So end to end, it was device up to IoT Core, which gets bridged into Cloud Pub/Sub. Pub/Sub had a Cloud Function that pulls from it, shoves it into Cloud Firestore. Firestore triggered with its callback into an angular app.

MELANIE: Nice. Yeah.

MARK: Wow.

GABE: It was pretty cool.

MARK: It all worked three times quickly.

GABE: I know, I can't. I did it once. That was good enough, right?

MARK: That works.

MELANIE: That was really good.

MARK: If people-- is there like a public sample or anything that people can actually go and have a look at?

GABE: Yes. So the easiest way to find it is going to be if on YouTube you do a search for "building iOS applications on Google Cloud." The only one on there. You'll find our video from the conference, and in the links in the description is linked to the GitHub with all of our code.

MARK: Awesome.

MELANIE: Well, great. Well, thanks, Gabe. We appreciate you coming on and helping us do the question of the week.

GABE: Thanks for having me.

MELANIE: All right, Mark, where are you going to be?

MARK: What am I doing?

MELANIE: Whatcha doing? Whatcha know?

MARK: So when this comes out, I will be in Tokyo--

MELANIE: Yes, you will.

MARK: --getting ready for Tokyo Next. So I'm pretty excited about that. Going to be talking there about iguanas, and talking to some gaming people, and stuff. So if you're in town, come say hi. What are you up to?

MELANIE: And I will be in Stellenbosch. I think I've probably said that a lot at this point. But I will be there doing some interviews. And then you and I will both be in Strange Loop at the end of the month.

MARK: Yeah, which is going to be really good. Gabe, you going anywhere special in next month or so?

GABE: I am. I'm doing a European trip. I'm doing London, Paris, Barcelona for a series of conferences.

MELANIE: Oh, that sounds difficult.

GABE: Yeah.

MARK: Which conferences are you going to?

GABE: So London is Cloud Next--

MARK: Yeah.

GABE: --which is our kind of next iteration of our series. And then Paris is visiting teammates because I have a weird kind of five days I don't want to fly back and forth to Europe. And the Barcelona is the big IoT conference, IoT Solutions World Congress, which is kind of the big European IoT conference.

MELANIE: Nice. Well, have a good trip.

GABE: Thanks.

MELANIE: Mark, I think that's it for us this week.

MARK: Yes. So Melanie and Gabe, thanks for joining us for yet another week on the podcast.

MELANIE: Thank you.

GABE: Thanks.

MARK: And thank you for listening, and we'll see you all next week.

[MUSIC PLAYING]

Hosts

Mark Mandel and Melanie Warrick

Continue the conversation

Leave us a comment on Reddit